

QLUB

News for the QL user

Issue No. 5

MAY/JUNE 1985

QL ARRIVES STATESIDE

You will soon be able to send your US penpals messages on Microdrive. The official stateside launch of your favourite micro takes place this month, with UK managing director Nigel Searle moving back across the water to supervise the operation.

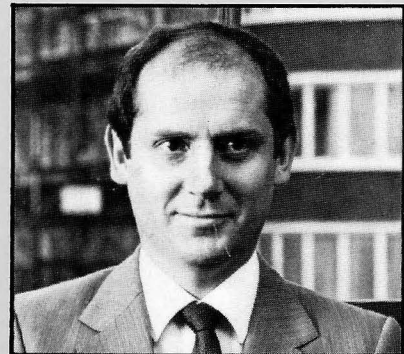
Dr. Searle, who previously directed Sinclair's US operations in selling ZX-80 and ZX-81 computers, is optimistic that the QL will find a comfortable niche in the admitted competitive US market. He points out that more than 250,000 ZX-80 and ZX-81s were sold in 1980 and 1981 before Timex started producing the computers under license—and that the QL will offer a lot more for the money than any other machine currently in the US home market.

Dr. Searle has also said he believes there are many US hobbyist computer users who have been left behind by the moves Apple

and IBM have made in an effort to capture the professional market. For both these hobbyists and budget-minded professionals, the QL will provide a cost-effective alternative.

To help reach these customers, Sinclair Research has entered into a joint distribution promotion with American Express—whose US cardholders will be among the first invited to buy a US QL.

So if you want to correspond with QL Quill files instead of plain old-technology paper, tell your US friends to be on the lookout for the QL.



Nigel Searle, former UK managing director of Sinclair Research, is now directing the US launch of the QL.

INSIDE QLUB NEWS

QLetters	2
QLibrary	3
QL to IBM	5
C previewed	6
Quantum Leaps	8
Competition	9

QL FUN AND FINANCE

The software featured in last issue's special section was only a fraction of what's now on offer. Software new this issue includes QL Home Finance—a package to help you make the most of your pay packet, QL Entrepreneur—software to help you start up your own business when you feel like telling your boss to pack it in, QL Cavern—a game to sharpen your wits, and QL Monitor—a programming tool that will help you test your own software.

QL Home Finance, which will sell in the shops for £24.95, is a powerful and flexible domestic finance program. It lets you rapidly enter payments and receipts into the Home Finance system, and all account balances affected are updated immediately. Even standing orders can be entered, and

they will take effect automatically when their due date arrives.

QL Entrepreneur is another classic business planning program from Triptych Publishing Ltd. Its RSP is £39.95. The program's primary intention is to provide you with the facilities to perform a detailed financial analysis of your business plan. And if you are unfamiliar with business planning and financial practises, then the program's built-in teaching section will provide you with the necessary background.

QL Cavern will provide you with a break from all that heavy thinking as you use keyboard and/or joystick to guide yourself to underground fortune (and possible fame). It's a good chance to see how much

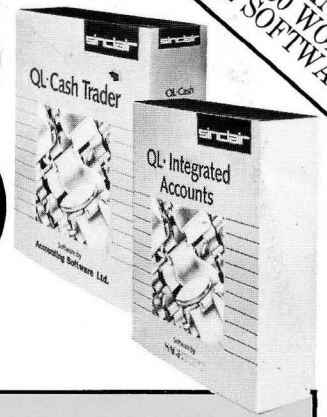
fun your QL can be! The 50-screen game will be in the shops later this month for £12.95.

QL Monitor, however, requires a very different kind of brainpower. It's a powerful program development tool incorporating machine code monitor, disassembler and line by line assembler, allowing you to (among other things) display and modify 68000 registers, single-step and run programs with multiple break points and develop special display windows that do not overwrite existing screen images—RSP £24.95.

All Sinclair software titles are available from Sinclair Research, Stanhope Road, Camberley or by telephoning 0276 685311.

sinclair

EXTRA
INSIDE
YOUR CHANCE TO
WIN £100 WORTH
OF QL SOFTWARE



QLUB grows up

There comes a time in the life of every club—whether its a C-L-U-B or a Q-L-U-B—that the membership starts to look for ways of improving it. Well, as you saw in the last issue, we've started that by offering occasional pull-out specials and extending QLUB membership to more people.

Now that we've got a expanded membership base, we'd like to serve all of you better. The membership expansion itself accomplishes part of that as there's far more of you to support each other in using the QL.

In addition to providing information about new products and programs, we at the QLUB Newsletter would also like to be a conduit for information between you. If you've got a routine or program you've developed or just an effective way of hooking

up your hardware, we'd like to know about it.

Being a member of QLUB doesn't just mean getting discounts and updates of software, it means participating in using the QL—for fun, for business and entertainment.

As you're all probably aware, a home computer (and particularly the QL) is unlike any other appliance you may have in the home. It isn't a single-purpose thing like a stereo, it isn't a passive thing like a television and it isn't as immediately utilitarian as say, a fridge. A computer has a vast number of uses and they're limited only by your imagination.

When you bought your QL, you received four applications packages with it, plus the SuperBasic programming language built in-

to the machine. But those were just a start. Obviously, they're useful and still among the best applications packages you can get—but they were a start, not an end in themselves.

WELCOME TO THE QLUB

THIS is the first of six newsletters that you will receive over the next twelve months.

Psion Software Support Limited also supply a comprehensive support service on QL Abacus, Archive, Easel & Quill, Qdos, SuperBASIC and any related peripherals — eg. Printers or memory expansion boards. Psion may be contacted by telephone on 01-723 0553 or by writing to:

Psion Software Support
22 Dorset Square
LONDON NW1

Q LETTERS

QL owners now and then

I am writing to you on behalf of those of us who have already obtained a QL and those who intend to in the near future.

The QL was very much ahead of the times (sic) when it was launched. Even if the same specification had been kept as in the launch model, it would still have been outdated by Atari's new S.T. range. Unfortunately the final versions were required to have a lower specification.

From a survey carried out by us over a wide range of computer users, the results indicate the features that people most wanted were:

a) Microdrives — there are two possibilities here —

i) Reliability must be improved at all costs. We know that this is possible due to having reviewed the One Per Desk and its drives.

ii) A 3 inch micro disk could be included as standard. If this is not possible, an interface at least should be included for disk drives.

b) The non-availability of software has indeed put off many a potential QL buyer. A very good example is set by Commodore's new C128 computer i.e. the compatibility with existing CBM64 software. Obviously a Spectrum emulator with cassette interface would be a great asset. It would not make business sense if the user then began to rely solely on cassettes and not use the Microdrives. Due to this factor, perhaps a Read

Only device would be more acceptable.

c) AS Qdos was never intended to be the operating system that would be marketed, it should either be replaced or updated to a better one which would incorporate multi-tasking in all modes including basic.

d) All the four packages should be incorporated in ROM and should be more reliable.

e) Super BASIC execution speed should be improved.

Our sources indicate that you already plan to make a QL+ (pardon the pun) with some of these enhancements already, so these improvements, with a welcome increase in RAM, should be very simple to implement.

I beg of you not to leave existing QL users stranded. Please reply.

QL Owners Society,
Warley,
West Mids.

First of all, there's no way the QL will be 'outdated' by Atari's new computers — if and when they finally arrive in the shops. The QL is a one-year old machine which is heavily supported by us and by third-party software and hardware manufacturers to offer a wide range of user options. Thus far, the unseen ST machines are not.

Secondly, the Microdrives — which we admit had initial teething problems — are now working reliably and well as you yourself note in your comments about their operation on the OPD. ICL has used a different — and slower — operating system on the OPD which would not be appropriate (or necessa-

ry) for the QL.

If you're lacking software, just look at all the new software announcement in last issue's Software Update and this issue's news stories. You'll find no shortage of quality programs. As for the Spectrum emulator, the end-cost of such a device would likely exceed the current market price of a Spectrum and as such it isn't worth considering.

As for disk drives, a growing number of disk interfaces are available for the QL — including products from CST, Quest and MicroPeripherals. And these interfaces will operate with 3.5 inch disk drives — with capacities of more than 1.5 Megabytes for a dual drive system.

It is simply not true that Qdos was 'never intended' to be the QL's operating system. Qdos will accomplish most of the tasks you're likely to require of it in day to day operation. However, if you are interested in other operating systems, they are now becoming available (including CP/M 68K from Quest, 68K/OS from GST).

ROM versions of the Psion software would unnecessarily add to the price of the QL — which we feel is one of the best computer buys on the market. If you would like Xchange in interior or exterior ROM form, what would you feel would be a reasonable charge?

QL defender down under

I feel bound to write to you, giving my basic impressions of the QL which I received as per details overleaf. I was disgusted with the review of the machine in the 'Australian

Personal Computer' last month. It was too much to see further denigration in a magazine using the Sinclair name, presumably to get a large circulation.

I suppose that any publicity is better than none, rather reminiscent of the chemist who made a small fortune out of marketing the much maligned 'Brand X' detergent – a lot of people were willing to try it out, presumably to prove it wasn't that bad.

I can understand that people were disappointed at the almost inevitable delay in getting the machine marketable. It was to their advantage finally. There was a warning too. I heeded it and I was pleasantly surprised that my UK colleague got delivery within 3 weeks of the time limit you quoted.

My main problem since its arrival has been to get hold of a suitable high resolution colour monitor. I paid a deposit for what was quoted as a high resolution, only to find a medium resolution was forwarded to the local Computerland store. I need it for a special music display system which I have worked on since the mid-sixties, involving the reading of music by scrolling notation upwards to a datum line resembling a piano keyboard, using slow scrolling techniques, coloured 'staff' lines spaced to allow for 'key changing' by simple panning; it also involves the use of 2 to 8 for white notes and AE IOU to represent the black notes to enable 'chord symbols' to be written accurately, thus enabling chord progressions to be easily interpreted, a case of finally applying logic to music notation, something it has

sadly lacked since its inception.

The only high resolution monitor available is suitable for IBM. This has a positive-going sync. I will have to resort to importing one privately as the final resort. Even with monochrome, it appears that insufficient details are given for the QL video output; I am asked to specify the vertical and horizontal sync. pulse frequency and whether the display is interlaced. In the UK this may not be difficult to resolve; you mention an international standard for monochrome in the manual. Here they maintain there is no such standard. The tiny Apple Macintosh is certainly non-standard. Two phone calls to Camberley on Friday failed to elicit the required information, promised now for tomorrow.

With only one TV available, my time on the QL has been somewhat restricted, to put it mildly, but I have noticed that Abacus does not mention the 'overwrite' option when saving an updated spreadsheet. Also the sum (col) function doesn't work on the early examples given; I spent quite some time trying to spot my non-mistake. Finally I found the answer in the simpler form: sum (N1:N15) syntax. I still need to find out how a 'bank balance' type of spreadsheet can be set up to enable credits and/or debits to be automatically reflected in the next column to the right – I ran out of time and patience on that one. The Microdrives have worked excellently up to date, but there is insufficient information on using 'backup' in Abacus. I got into a loop with both Microdrives taking turns incessantly, possibly be-

cause the spreadsheet was still in memory. I risked using reset, finally getting out of it, presumably, by wiping the program from memory. What is the correct procedure?

I feel quite certain that the 'knockers' will have to eat their words when the QL is compared with other machines. My son would have been the proud owner of a Spectrum from last Xmas but 3 supplied to the local K-Mart Store were all colourless. Barsons, your importers, failed to repair them. They are still there, at a reduced price, sad to say.

I wish that this Praxis 45D was hooked up to the QL. The price asked for the interface however is \$490 plus fitting costs, so I prefer to wait than be 'mugged' to that extent. A twelve inch colour monitor for the IBM due in mid November is a staggering \$1240A.

Forgive them, they know not how lucky they are over there.

M. Hughes,
Nambour, QLD,
Australia.

Readers – any advice you can give to our antipodean friend would be appreciated.

Software Specials

To order DIRECT FROM SINCLAIR RESEARCH Ltd: By mail: complete the coupon below. By telephone: please phone 0276 685311 9am–5pm, Monday–Friday (remember to have your credit number at hand).

The following titles are available at a 20% discount from this issue of QLUB: QL Monitor (QLUB price £19.95), QL Toolkit (QLUB price £19.95), and QL Assembler (QLUB price £31.95). Other software immediately available from Sinclair Research include QL Chess, QL Touch 'n' Go and QL Cash Trader.

Qty	Item	Item Price	Total

*Please delete/complete as applicable Please tick if you require a VAT receipt ☐

*I enclose a cheque/postal order payable to Sinclair Research Ltd. for £ _____

*Please charge to my Access/Barclaycard/Trustcard No. _____

Signature _____

£ _____

PLEASE PRINT

Name: Mr/Mrs/Miss _____

Address _____

QLB005

Send to Sinclair Research Ltd., FREEPOST, Camberley, Surrey GU15 3BR

INTERFACE INQUEST

QL and the Monitor

To get the best picture out of the QL, you'd be well advised to use a colour or monochrome monitor. The QL is a colour computer and therefore to obtain the best results from most applications you need a colour monitor. However, a good monochrome monitor is better than a bad colour television if you don't really need to distinguish different hues.

Sinclair Research offers its own colour monitor, the QL Vision Colour Monitor, featured in QLUB issue No. 2 as the prize in our contest—but choosing a monochrome monitor involves a little bit more thought and care, as there are many on the market and none specifically sponsored by Sinclair Research.

The thing you'll be looking for is clarity of picture, low glare on the screen and a high resolution of the characters. Because the QL has a unique 85 character screen, you'll find that some monitors won't properly display all that the QL has to offer. So, in looking for a monitor, you should see first of all the density of the characters and whether or not the monitor can display the 85 characters width. Having done that, and of course found a monitor that is in your price bracket, you're ready to think about hooking it up.

Most monochrome monitors use what's known as a standard phono plug at the monitor end of things, while the QL itself uses a multipin din plug to send out its monitor signal. As there are only two lines for the signal at the monitor end, you'll have to pick the correct two lines from the DIN plug end to connect the monitor to the QL. Those are pins numbered 1 and pins numbered 2 as shown in figure 1. Pin 2 is the grounding pin, while pin 1 is the main signal pin.

You can either buy this lead from any one of a number of shops or make your own. What you'll need is an 8 pin DIN plug and a standard phono lead—which you should be able to get from any hi-fi shop or specialist electronics shop. Solder the ground end of the hi-fi lead to pin 1 and the video signal end to pin 2. Having done that, you should be ready to hook up your QL to the monitor with a reliable connection which should give you a clear and well-defined picture.

When using the monochrome monitor you'll have to keep in mind that the QL is a

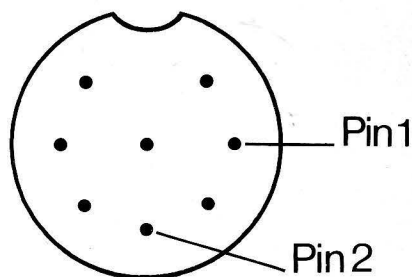


Fig.1

colour computer and therefore the shades that you'll get and the definition you'll get will depend on what colours you're using with your programs.

Although you can't change the colours in the programs that come with the QL, QL Abacus, Archive, Easel and Quill, you can control the colours used in your own programs and utilities. Obviously if you are using a monochrome monitor, you'll want to make sure those are the starkest contrasts for easier reading, so that perhaps instead of red and blue which are difficult to read together, you'll be looking at black and white, or perhaps black and yellow as white tends to send off a bit more glare.

The other main advantage of using a monochrome over a colour monitor is one of size. Monochrome screen technology has been around longer than colour TV tube technology and miniaturization as epitomized by the Sinclair pocket TV for black and white technology is more advanced.

You'll find that monitors such as the Apple IIc monitor—which Apple markets for its IIc portable computer—are well suited to the QL's small size and can be used effectively with it. The IIc monitor, however, is slightly more expensive than most of the others you'd find on the market, even though it is smaller and fits in with the QL's small desktop footprint.

COMMUNICATIONS CORNER

QL to IBM PC

Your QL is equipped with two serial interfaces for communications. These allow you to send information to printers and to receive information through modems and directly to/from other computers through the RS232 connection.

The IBM PC is no exception to this and because so many people have IBM PCs in their offices, the ability to transfer files to and from the QL is particularly useful.

The IBM PC doesn't come standard with any serial interfaces - although most memory expansion cards and many IBM PC compatibles do include a serial interface, which is labelled as COM1 by the PC's operating system. In order to communicate between QL and PC, you'll need an IBM PC or compatible with a serial interface and a communications program, a female-to-female connector going out of the back of the IBM PC and the standard QL serial interface cable. The standard QL serial interface cable is connected to the female-to-female RS232 connector at the back of the IBM PC which then goes in the RS232 pin slot in the IBM. The types of files you can send are largely limited to text files as program files would be useless on each of the machines.

IBM PC programs can't run on the QL and QL programs can't run on the IBM. Only in certain specific cases, where for example the IBM PC was running PSION's Exchange program and the QL was running one of the four PSION applications packages, could you properly exchange non-text files.

Whether you plug the serial interface cable in the QL's serial port 1 or serial port 2 depends on the connections at the IBM end. For most applications, you'll plug it into serial port 1, although an easy way to test is to try either port and obviously use the one that works.

To test the connection, load up your IBM PC's terminal program and adjust the settings to the following: 300 baud, even parity, 1 stop bit, 8 data bits, no echo and no line feeds and Xon, Xoff in the off position. With these protocol settings in place, you now type the command line on the QL;

COPY CON. TO SER1e

if you've got the QL connected at serial port 1 (type SER2e if you've got it connected to serial port 2).

Now, anything you type should appear on the screen of the IBM PC. When this happens, you'll know you have the transmission connection connected alright. What you need to do now is to test whether you have the receiving connections wired up correctly. To test that, you must hit the control key and the space bar on the QL to break the QL's command - which is currently waiting for information to be sent to the IBM. Now type;

COPY SER1e TO SCR.

and hit return. Now try typing something on the IBM and it should appear on your QL screen. When it's done so, you have the two machines connected up correctly and you can send and receive files between them.

As we said before, in most cases you'll be dealing with simple text files (ie documents that you've written on one machine and would like to manipulate on the other.) So far what you've seen is how to copy information for one machine's keyboard to the other machine's screen. What you want to do is copy from one machine's disk drive to the other machine's Microdrive or disk storage system. In order to do that, you simply change the name of the device that the information is being sent to. The QL's operating system-QDOS-is operated on a principle known as device independence which means you can quite happily send information from the PC to the QL's Microdrives in the same way you can send it from the PC's keyboard to the QL's screen.

In order to set up a file in which to receive, all you need to do, assuming that the baud rates and the communications settings are the same on the two machines is type:

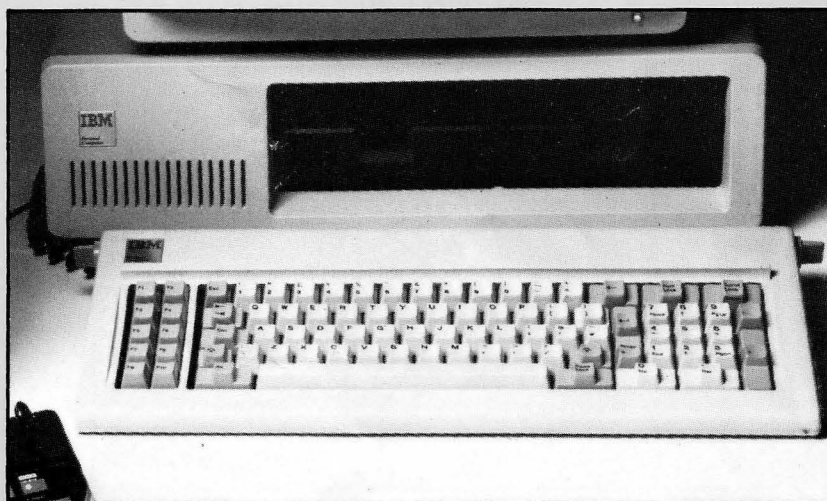
COPY SER1e to MDV2.(the name of your file) .EXP

What this does is sets up an export file on Microdrive 2 which can then be read into QUILL if you wish to do so after communicating. You can then send the file to the IBM in one of two ways: if the communications program on the IBM allows for the simple sending of text files, use the send command and send the file in the normal way to the QL. You'll see it being received as the Microdrive accesses to save the information as it comes in.

If your communications program will only transmit with the Xon, Xoff protocols, and transmit as program files, you need only enter PC DOS, set the baud rate to 300 baud and the parity to even, using the mode command in PC DOS, and then type the phrase:

TYPE (NAME OF FILE HERE)>COM1:

This will then cause the text file you want to send to be typed to the serial port in the same way the QL copies. The COPY command on the QL is very similar to the TYPE command on the IBM PC.



New ways of C-ing

What is C? That's simple...it's one of the most highly-regarded of the more modern high-level computer-languages. Alright, then more precisely, why is C any more interesting than other languages?

The answer to that one is quite simple too. It's popular among working programmers because it makes their jobs much easier. And it's how it does it that makes it interesting.

Most computer-languages are the result of long deliberations by learned committees, with the result that like a mouse designed by a group, they come out looking like camels. C, on the other hand, was developed by a couple of individualists in a back-room on an unwanted (and not very big) PDP-10.

These two, Brian Kernighan and Denis Ritchie, had been working on a very big, complex and costly system called Multics, and their experiences had led them to form the opinion that the main problem of such projects was that the facilities available to the programmers, were, all too frequently, less than helpful, and seldom accurately represented the capacities of the machine being programmed.

What was needed, they felt, was a new system; one which helped the programmer as much as possible, yet which didn't impose too many restrictions on his or her options.

This is in direct contrast to the philosophy underlying such languages as Pascal, which was developed at about the same time, and which assumes that programmers, (being human) are liable to make mistakes, and so should be constrained as far as possible into paths which can't cause too much harm.

Junking this idea, and providing plenty of low-level operations with as few constraints upon their use as possible was a better idea, they decided.

The result is that C has 35 distinct operators...more than any other language except APL (which doesn't have anything else!). And since C compilers operate on the assumption that whatever it's being asked to deal with makes some kind of sense (even if it looks horribly dangerous), if it's syntactically correct, it'll produce some kind of code.

However, it does mean that you can produce some really spectacular system-crashes, but it also means that you can treat things any way you want...you won't be told that what you want is illegal, when you know precisely what you're doing. However, it's much (MUCH) easier to write complicated stuff, like compilers, spreadsheets, editors and the like. So much so, that many programs which we know and love (Wordstar, VisiCalc, and even BASIC

itself) are being re-written in C, simply because it's so much easier to maintain and upgrade than Assembly-language.

The wide-ranging powers of pointers in C are the most important consequence of this lack of limits. Pointers are variables which don't actually have a meaningful value themselves, but which contain the address of, and thus 'point' at another variable which does. In fact, it's fair to say that C and pointers are virtually synonymous; certainly without them C wouldn't be much to write home about!

But when coupled with the data-organising facilities of 'structures', which allow groups of dissimilar items to be treated as individual entities, and 'unions' which let a programmer keep more than one kind of data in the same place, the pointer becomes a kind of 'magic-bullet' which can tackle and overwhelm even the most convoluted of problems. And all because C doesn't try to figure out where the pointer 'ought' to be...it's honest enough to admit that maybe programmers aren't complete fools, and presumably know what they're doing.

There are only 27 'reserved words' much less than in BASIC, and of those, 14 are concerned with variables - what kind they are, how big, and where they're kept. The remainder provide a reasonably clean set of phrases, namely a 'for-loop', a 'while-do loop' with matching 'do-while loop', and an 'if-clause' with an optional 'else' which can often be replaced with a 'switch-case-default' block. Any situations which can't be handled with these can be mastered with a 'labelled-goto', a 'break', or a 'continue', with values being transferrable by means of a 'return'.

No mention of any words that read or print, though...does that mean you can't get anything in or out? Not at all, but C doesn't have *any* built-in Input or Output. Why not? Well, the problem lies in the fact that all machines differ in how they actually work, so Kernighan and Ritchie decided that the best way of controlling the actual hardware, which is the only thing that differs greatly from machine to machine, was to leave it entirely up to the programmer!

This makes more sense than might appear at first, because everything which isn't an operator or a variable in a C program has to be a function. So all I/O goes to or comes from one or other function. Hence another feature of C, the libraries, which can be expanded so that any conceivable task can be handled.

So there it is...C is a modern, flexible, expandable language which offers careful users the chance to tackle and master problems which would have defeated many other languages. Not a language for clumsy people, but which, not having to cater for them, isn't limited.

AFTER TAKING EVERYTHING UNSTRUCTURED LANGUAGES CAN THROW AT THEM, THE TATTERED REMAINS OF THE ONCE PROUD COMPANY REACHES THE "C".



First past the post in producing a version of C for the QL is GST. GST's 'QC' C compiler - available at a discount through QLUB - became available in February.

QC arrives in a large black and grey box which also contains a loose-leaf A5 ring-bound manual, a formidable-looking tome called 'A Book on C' by R.E. Berry and B.A.E. Meekings - which functions both as a tutorial and reference-manual for the language and, of course, a Jiffy-bag containing four Microdrive cartridges, (two with the various parts of the system, and two blanks).

There's a sound logical reason for including the Berry and Meekings book as part of the package. A large chunk of it consists of the listings of a Small-C compiler, from which QC is drawn...so actually, it's part of the system documentation.

That's really all the information you'll get about how things happen. Neither of the Microdrive cartridges contains any library source files, so unless you're prepared to totally replace a supplied function, you're rather stuck with the way it works.

Not that you'd often want to modify standard library routines, but you might, and since part of the philosophy behind C is that users should be able to alter anything they feel a need to (and moreover most C-systems do supply the source-code to the I/O library and so on) QC's lack is all the more noticeable.

To program in C, you'll also need some kind of text-editor, which you have to provide, and it would also be useful to have the QL Toolkit. However, if you do have the QL Toolkit, the system is greatly enhanced, and in some ways is as flexible as UNIX, often considered to be C's true home. The QL Toolkit allows the use of I/O redirection, pipes, command-line parameters and flags, and several other features.

The actual C-system consists of the compiler and its associated assembler, the Sinclair linker, two library-files, some control-files, a header-file and just one example-program, plus a SuperBASIC copy-program called CLONE.

The system will run on any size of QL, and all ROM versions. As mentioned, the QL Toolkit is useful but not essential. The Microdrives are quite satisfactory provided you don't write very large programs, but floppies would be an advantage if you plan on very big systems.

Surprisingly, in view of the fact that the 68000 processor was specifically designed to run advanced languages such as C, QC isn't the full-blown variety, but a version of Small-C. This dialect, which is tidier and has a few more features than the original Small-C by Ron Cain, but less than the Kernighan and Ritchie definition of real C, is also known as Rat-C, and hence the reason for the cut-down version.

Small-C was originally developed for the 8080 and Z-80, processors which are less than ideal for compilation, which is why the original author left out some of the more advanced, but most useful features of C, such as structures and unions, and their associated operators.

As a result, the most advanced data-handling mechanism which is inbuilt is the single-dimensional array...any more complex tasks have to be constructed from sets of small arrays or partitions of large ones. This means that many of the techniques so often employed in C programs, such as trees, linked lists, stacks, rings and queues, though possible to program, are neither as easy to construct, nor as simple to extend as might be wished.

Similarly, there are no floating-point values, nor doubles, but since everything, (including eight-bit characters) is operated upon as 32-bit quantities, this isn't quite as serious as first appears...anything that won't fit into a 31-bit value raised to a 31-bit power is more than any reasonable person could ever want, isn't it?

Rather less convenient is the pre-processors' inability to cope competently with 'include-files' containing other 'include-files'...this sub-section of the compiler runs over the text of the program before the compiler-proper, adding subfiles to the source and translating simple symbols which the programmer has defined for ease of reading into the proper C-syntax.

Creative use of the pre-processor can make programs easier to read and modify, as well as providing the simplest mechanism for breaking big programs into manageable pieces and this restriction, whilst not fatal, means that long lists of '#include' statements are a likelihood.

Similarly, the pre-processor can't take macros with parameters, so although you can define almost anything as a suitable candidate for translation, each different case will need its own macro.

In some ways, however, the 77 functions in the standard library supplied with QC are actually more capable than the ones on many real C systems...for example the string-formatting and conversion routines not only recognise all the standard formats, (signed and unsigned decimal, octal, hexadecimal, character and string-address), but binary as well. Two other useful but non-standard functions are provided for parameter-handling, one which makes functions with variable numbers of parameters much simpler to deal with, and the other, which interprets and handles the command-line arguments.

The graphics and sound-facilities of the QL are handled by a special Qdos library containing 47 functions ranging from the very low-level traps to some complicated graphics routines for drawing arcs, circles and ellipses, not to mention filling! Actually, these weren't available in the pre-release version, but they're promised for the final ones.

The one feature which makes QC really different, however, is the #asm directive...this isn't part of the language, as such, but a pre-processor command. The effect is to let you include 68000 assembly-language in your programs. Happily, sufficient information is provided concerning the use of the stack to make this seem straightforward. Anything you can't do in C, you can, therefore, perform directly on the machine.

Of course, any code is almost certainly machine-dependent, and totally compiler-specific, but with it, you should be able to make the QL do things which no-one ever expected. But isn't that what the pleasure of computers is supposed to be all about?



Quantum Leaps: The QL Case Study

(This is a copy of a letter one of our erstwhile Australian readers and one of our biggest antipodean boosters - M. Hughes - originally sent to Sinclair User. It is reprinted here at the behest of Mr Hughes, whose letter to QLUB appears on Page 2)

My QL arrived at the end of August, giving me a few weeks to ascertain its comparative value vis-a-vis the rest of them.

It may be worth explaining that I first saw the QL mentioned in 'The Australian' (he owns the Times etc.) at the end of January. The very brief description was sufficient to get a letter off to Sinclair. Delivery was made on 20th August (UK). I had to pay AU \$40 airmail costs, your VAT, then about \$110 in local duties, (based on full UK retail price) plus all postage costs. Easel had been 'wiped clean', presumably by Customs X-ray check. A replacement is since to hand.

The most obvious action to take after a quick trial of the machine, was to get another backup on order. Why? The price comparison with other machines and the knowledge that they all fail sooner or later makes it so much easier to duplicate the whole system; when one does fail, I will have not only a standby, but also a set of spare chips to enable me to pinpoint the problem, in most cases.

This is my first experience with any computer; I've regarded them as toys or very overpriced equipment, until recently; the QL was the exception, hence the plunge. I've built up a 7 figure overseas company over the last 25 years using ballpoint, paper and brains. A computer has never seemed a necessity. This still applies.

I was earning my living in electronics 50 years ago (30/- a week). That tells me you are very fortunate to have someone of the calibre of Sir Clive Sinclair to keep the UK on the leading edge of computer technology. If one reads 68000 Assembly Language Programming by Gerry Kane, Doug Hawkins and Lance Leventhal any comparison of 8 bit internal architecture chips becomes ridiculous. The 8 bit external bus limits the data transfer rate to that of the IBM PC, a much more expensive set-up than the QL. Programmes for all the 68000 family is acknowledged to be at an early stage of development. The wonderful part of the QL is the set of four software packages included in the 'crazy' price; these are being improved continually, and again, in contradistinction to other expensive packages, an update is available at very low

cost via QLUB. None of the four packages took 50 seconds to load, on my machine - buy him a new watch - Abacus, a really useful spreadsheet, took 30 seconds.

Access time is not the same as loading time; my 14 year old son informs me that disks at his school take considerably longer to load. I do know that the Union Jack on the Spectrum comprises a long list, and takes a long time to run, whereas some 25 lines sufficed for my boy to programme a similar one on the QL, and it takes shape in a fraction of the time, due to the machine code blocks such as 'fill'. I did have qualms about the Microdrives. To date they have operated well. I did get into a continuous loop trying to backup a file. I didn't have time to read through the voluminous manual. The reset button solved the problem. This does not appear in the book.

On a more practical note, using QL Abacus to determine the total charges on some Share Contracts, the percentages highlighted two errors. One, a failure to charge the 1% Stamp (\$234), subsequently debited. Two, stamp charged at 1½%, an overcharge of \$132, not yet credited, but now being reclaimed from Brokers. It wouldn't take long for the QL to earn its keep if such errors would otherwise go unnoticed. The real problem is that the QL is much too versatile to utilise fully in the small business field. Compare the graphics from this multi-billion dollar Company's Annual Report, with those available from the QL and you won't knock it.

M. Hughes,
Nambour,
Qld, Australia

(This letter is about an interesting application of the QL that's a little closer to home.)

Since August last year I have developed a strong admiration for the QL and the four software packages.

However, Archive, perhaps because it is the most elaborate, has always given me more trouble than the others when I have set it to work on my various applications.

So far, though, I have always managed in the end but, since the last occasion was especially trying, your readers may find it of interest. I should anyway be grateful for comment.

I use a Mannesman Tally SPIRIT 80 printer and wished to print a 'family tree' from a coded file of genealogical records, using condensed characters at 142 per line and with repeated use of the 'lprint at' command.

First problem - no installation programmes on the Archive tape, although the handbook implies that they are there.

I made a new Printerdat file using the Abacus copy but it didn't work. Then I tried all the correct chr codes in correctly formatted lprint instructions but they would not work except for ESC 3 which did modify the line spacing, albeit strangely. In spite of SI and ESC Q, the print size and page width, in columns, refused to budge from normal and 80.

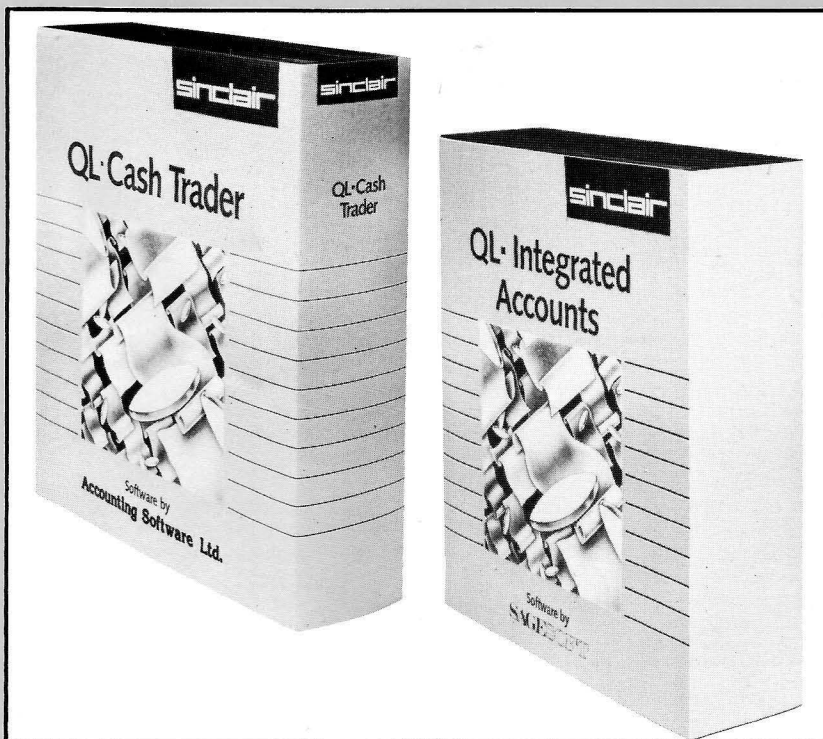
So I ran an analysis of all the ESC 2 codes with and without nuls and chr(27)s. As a result I can tell you that 'lprint chr(0); chr(27); chr(29)' works like the SI code when played through Archive whereas chr(15) with or without nuls does not.

Now I can do a really nice family tree but for a mere user it was a tough fight to get there. Still, that's Archive for you.

David Northwood,
Tunbridge Wells,
Kent.

Early versions of the User Guide did not make the printer installation procedure completely clear. In fact, you normally need to use the printer installation routines combined in QL Abacus or QL Quill to install a printer in QL Archive - a point that is now explained by the User Guide. The reason why the printer installation programs are not fully present on QL Archive is that there is insufficient room on the cartridge. If you want to provide them all on the same cartridge, you need to make space. The easiest way to do so is to copy the test file 'gazel' from QL Archive onto a spare cartridge, then delete 'gazel' from the QL Archive cartridge. Then you simply copy 'install_bas' - the missing file - from QL Abacus or QL Quill on to the QL Archive cartridge.

Your chance to win £100 worth of free QL software



Perhaps we make these competitions too easy – they didn't seem to deter many people from entering the contest held in Issue No 3 to pick 25 recipients of the QL Chess program. You'll see the winners listed in the column at right – but first a little about this issue's prize and how to win it.

With this issue you get a chance to win £100 worth of Sinclair QL software – your choice of titles – and all you have to do to qualify as a potential winner is answer a few simple questions. The first correct entry we draw will win. Entries must be in no later than July 15, 1985 to qualify and the winner's name will be published in the September/October edition.

Here are the questions;

- 1) The QL's Microdrives use a particular quality of magnetic tape for storage. What other medium is this tape used in?
a) Videotape b) Commercial sound recording c) Film recording
- 2) In order for the QL to load a program automatically, you must save a certain file onto the Microdrive. What must you call that file?

- 3) What recently-announced British Telecom business system also uses Sinclair Microdrives?

Send your answers along with your QLUB membership number (although, you shouldn't worry if you haven't received it yet) to:

QLUB competition
The Editor
QLUB News
Sinclair Research Ltd.
25 Willis Road
Cambridge CB1 2AQ

The 25 winners of a QL Chess program are:

MR C SMITH
STOCKTON-ON-TEES

MR E C ANSTEE
WENVOE

MR I MARTIN
HELSTON

MR M W S WHYMAN
GUISBOROUGH

DR D C BURNETT
GERRARDS CROSS

MR B DUCKWORTH
PUTNEY

MR R LAWRENCE
STANMORE DRIVE MIDDX

MR J AYRES
ILFORD

MR B GIBSON
BASINGSTOKE

MR G MATTHEWS
WEYBRIDGE

MR J TRICKER
KENTON, STOWMARKET

MR G P MILNER-WALKER
PETERSFIELD

MR W A BAVE
FARNHAM

MR N S TODD
LONDON W2

MR P J RIDOUT
WINNERSH, WOKINGHAM

MR E J SARGEANT
LONG EATON NOTTINGHAM

MR D S HAY
ABERDEEN

MR W TYRELL
NEW MILTON

MR MELJERAAN
LONDON W3

MR M O PENNEY
WORMLEY, GODALMING

MR C J JEFFERIES
YALTON, BRISTOL

MR G F COOK
HAILSHAM

MR D J PALMER
ST ALBANS

MR D KELLY
EPPING

MR I C J WOOD
STOCKPORT

Congratulations to all the winners and thanks to all who entered.

```

10 BAUD 9600
20 CLS:INPUT "Filename ";A$
30 OPEN #5,"MDVL_"&A$&"_EXP"
40 OPEN #6,SER1
50 A$=INKEY$(#6)
60 IF A$="" THEN GOTO 50
70 PRINT #5,A$;
80 IF CODE(A$)=26 THEN:PRINT "END OF FILE":CLOSE #5:STOP
90 GOTO 50

```


QLibrary

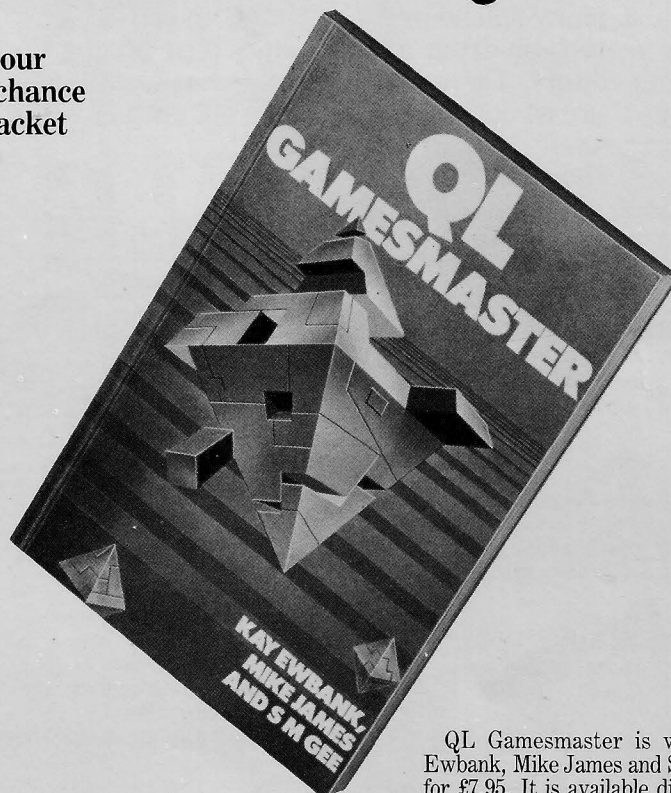
Welcome back to the QLibrary, where we feature new books on your favourite micro and give you a chance to see more than just the dust-jacket before you buy.

THIS issue we feature two books from both ends of the spectrum (or the QL for that matter)—one on gaming and the other on machine code programming. The first is QL Gamesmaster from Collins, while the other is Quick QL Machine Language from Melbourne House.

QL Gamesmaster is a well presented compromise between the basic programmers guide and a book of advanced programs.

Within its 175 pages the book uses clear, easy to follow steps to teach the inexperienced programmer the skills of formulating games programs (which themselves combine all aspects of programming). All the programs incorporate animated graphics and a series of type-in modules which are later strung together to form more complex programs.

The programs must be studied carefully as they are typed in and through an understanding of the way in which the modules are combined, you get a good picture of the best methods for putting games together.



QL Gamesmaster is written by Kay Ewbank, Mike James and S M Gee and sells for £7.95. It is available directly from Collins or any bookshop that stocks Collins books.

Quick QL Machine Language by Alan Giles is published by Melbourne House and is designed to provide QL users with a firm background in programming the 68000 family of processors—of which the QL's 68008 is a part.

Included in the book is an Assembler/Disassembler listing to aid you in machine code programming. The author states the purpose of these listings—and the book in general—in its first couple of paragraphs;

"I set out to teach myself machine code for the 68008, the processor at the heart of the QL, and to write an assembler to enable the development of machine code programs. On the way to achieving this aim, I wrote a disassembler which, by letting you look at the workings of the QL's Qdos and Super BASIC ROM set can help in the understanding of both 6800 machine code and the QL. Both the assembler and disassembler are written in Super BASIC and are presented in full in this book so you can type them in yourself."

This book—as we said at the outset—is an excellent background to machine code—but don't make it your starting point. It is not a book for beginners. Quick QL Machine Language sells for £7.95 and is available directly from Melbourne House or any bookshop that stocks Melbourne House books.



QL Quarks

Humour is a many-splendored thing and its power goes beyond the mere hum-drum and deadlines that plague the likes of news editors. There is no such thing as a stale joke (well, maybe there is, but we wouldn't dare tell one here) or a second-hand joke.

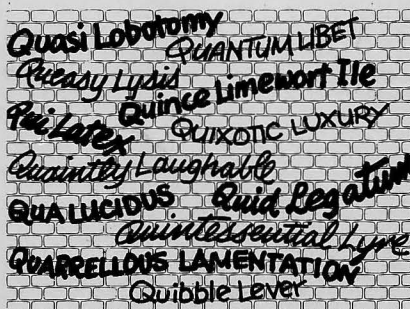
It is in this spirit that we present you with further limericks which didn't quite make the grade in our limerick competition. We thought it ungracious not to share with you some of the rib-tickling wit we received simply because a mere deadline had passed.

To wit, the following is offered for your perusal;

A QL computer from Cambridge
Once LOADED a file with the name
'Bridge',

And when ordered to RUN,
PRINTed (CSIZE 3,1),
'Suspension, or teeth, or the game,
bridge?'

B. Morgan,
Vienna, Austria



A QL computer from Cambridge
Met a Beeb in a train crossing the
Tamar Bridge
They interfaced at first station
Now the Next Generation
Seek Artificial Amour on a Seine
bridge

Fr. J.W. Harden
Southend

A QL computer from Cambridge
Was programmed by pert Percy Bain-
bridge
Compiling his Qdos with Pascal, the
rascal
Interfaced Unix at Uxbridge

C. Spencer
Burntisland
Fife

A QL computer from Cambridge
was bought by the Archers of Am-
bridge
and very soon Phil,
with Abacus and Quill
found the Archives were easily ma-
naged.

OR

A QL computer from Cambridge
could record on a Microdrive cartridge
all the menus you need
to suitably feed
a tribe like the Archers of Ambridge

(I regret the obsession with the Ar-
chers, but there are not many suitable
rhymes for Cambridge—couldn't you
move somewhere else?)

T. Rendall Davies
Covent Garden
London

SPECIAL OFFER OFFICE

The Special Offer Office keeps track of all deals and discounts for QLUB members—which includes anybody who is sent this newsletter. Listed below are the deals offered to members;

GST Computer Systems is offering its acclaimed 68K/OS operating system for the QL at an attractive discount. For details, write to GST Computer Systems, 8 Green St., Willingham, Cambridge CB4 5JA or telephone Jane Pate-man on (0954) 81991. Have your membership number and address to hand.

A 20 per cent discount is available on QL software sold by Sinclair Research through QLUB. It includes;
QL Monitor—QLUB price £19.95
QL Assembler—QLUB price £31.95
QL Toolkit—QLUB price £19.95

Send your order to Sinclair Research, Stanhope Road, Camberley, Surrey or telephone on (0276) 685311.

A discount is available for a year's subscription to EMAP's QL User magazine. A full year's subscription including delivery would normally cost £15. If you subscribe through QLUB, the magazine will be delivered for an all-in (postage included) price of £11.50. To receive your 12 issues, send a cheque or money order for £11.50 to PRQL, Subscription Department, QL User, Priory Court, 30/32 Farrindon Lane, London EC1.

NEXT ISSUE

- Hardware update
- QL games
- Metacomco software profile
- Focus on printers